Método de Gauss-Jordan

Resolver el siguiente sistema de ecuaciones por el método de Gauss-Jordan:

1.2	2.1	-1.1	4.0	5.98
-1.1	2.0	3.1	4.9	3.89
-2.1	2.2	3.7	16.0	12.2
-1.0	-2.3	4.7	12.0	4.03

Este método debe su nombre a Carl Friedrich Gauss y a Wilhelm jordan. Se trata de una serie de algoritmos del algebra lineal para determinar los resultados de un sistema de ecuaciones lineales y así hallar matrices e inversas. El sistema de Gauss se utiliza para resolver un sistema de ecuaciones y obtener las soluciones por medio de la reducción del sistema dado a otro que sea equivalente en el cual cada una de las ecuaciones tendrá una incógnita menos que la anterior. La matriz que resulta de este proceso lleva el nombre que se conoce como forma escalonada.

Command Window

```
A =

    1.2000    2.1000   -1.1000    4.0000
   -1.1000    2.0000    3.1000    4.9000
   -2.1000    2.2000    3.7000   16.0000
   -1.0000   -2.3000    4.7000   12.0000

Agregue la columna de resultados R[1]:  5.98
Agregue la columna de resultados R[2]:  3.89
Agregue la columna de resultados R[3]:  12.2
Agregue la columna de resultados R[4]:  4.03

B =

    1.2000    2.1000   -1.1000    4.0000    5.9800
   -1.1000    2.0000    3.1000    4.9000    3.8900
   -2.1000    2.2000    3.7000   16.0000   12.2000
   -1.0000   -2.3000    4.7000   12.0000    4.0300
```

```
Command Window

    -0.5901


B =

     1.0000         0         0         0    0.0880
          0    1.0000         0         0    1.0214
          0         0    1.0000         0   -0.5901
          0         0         0    1.0000    0.7701


La variable X[4] es:

ans =

     0.7701


B =

     1.0000         0         0         0    0.0880
          0    1.0000         0         0    1.0214
          0         0    1.0000         0   -0.5901
          0         0         0    1.0000    0.7701
```

Código utilizado:

```
clear all;
clc;
fprintf('RESOLUCION DE ECUACIONES LINEALES POR EL METODO GAUSS-
JORDAN\n\n');
m=input('determine amplitud de la matriz cuadrada ');
n=input('Cuantas incognitas ?enen las ecuaciones: ');
for i=1:m
    for j=1:n
        fprintf('El elemento A[%d %d]',i,j);
        A(i,j)= input (' ')
    end
end
A
for j=1:m
    fprintf('Agregue la columna de resultados R[%d]: ',j);
    R(j)= input(' ');
end
B=[A R']

for i=1:m
    B(i,:)=B(i,:)/B(i,i);
    for j=i+1:m
```

```
        B(j,:)=B(j,:)-B(i,:)*B(j,i);
        j=j+1;
        B
    end
    i=i+1;
    B
end
fprintf('HOLA HAREMOS CERO LOS ELEMENTOS INFERIORES')

for i=m:-1:2
    for j=i-1:-1:1
        B(j,:)=B(j,:)-B(i,:)*B(j,i);
        j=j-1;
        B
    end
    i=i-1;
    B
end

for i=1:m
    for j=n+1
        fprintf('\nLa variable X[%d] es:\n',i);
        B(i,j)
        B
    end
end
```

Conclusiones:

Este método, permite resolver hasta 20 ecuaciones simultáneas. Lo que lo diferencia del método Gaussiano es que cuando es eliminada una incógnita, se eliminará de todas las ecuaciones restantes, o sea, las que anteceden a la ecuación principal, así como de las que la siguen a continuación. De esta manera el paso de eliminación forma una matriz identidad en vez de una matriz triangular. No es necesario entonces utilizar la sustitución hacia atrás para conseguir la solución.

Sanchez González Oscar Eduardo.

ingeniería en Sistemas Computacionales.

Métodos numéricos

NUA:304987